# <https://github.com/nareshbunty/coding-helper.git>

# C:\Users\NARESH\git\coding-helper

# <https://eclipsesource.com/blogs/tutorials/egit-tutorial/>

# https://maxrohde.com/2012/05/25/eclipse-and-github-tutorial/

# https://guides.github.com/activities/hello-world/

# <http://www.baeldung.com/hibernate-4-spring> xml,beans both

# <http://javabycode.com/build-tools/maven/add-oracle-jdbc-driver-maven.html>

# <http://www.javainterviewpoint.com/spring-boot-deployable-war/> simply superb bvn

# http://javabycode.com/spring-framework-tutorial/spring-boot-tutorial/spring-boot-deploy-war-file-example.html

# <https://docs.spring.io/spring-boot/docs/current/reference/html/howto-traditional-deployment.html>

# <https://www.boraji.com/how-to-create-a-web-project-using-maven-in-eclipse#maven-run-tomcat> bvn

# [How to check java version of a running java process used](https://stackoverflow.com/questions/34591028/how-to-check-java-version-of-a-running-java-process-used)

Jenkins:

java -jar jenkins.war

How to change the default port in Jenkins

Start Jenkins from cmd line with this command :

java -jar jenkins.war --httpPort=8081

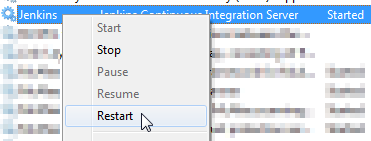
or

1. Go to the directory where you installed the Jenkins (In default, it is under Program Files/Jenkins)
2. Open the Jenkins.xml
3. Search --httpPort=8080 and replace the 8080 with the new port number that you wish

### Restart Jenkins service

Just **restart** the **Jenkins service** after you changed the port in **jenkins.xml**.

1. Press Win + R
2. Type "services.msc"
3. Right click on the "Jenkins" line > Restart

[](https://i.stack.imgur.com/jjJjo.png)

1. Type http://localhost:8081/ in your browser to test the change.

Use jcmd:

jcmd process\_id VM.version

$ jcmd 9619 VM.version

9619:

Java HotSpot(TM) 64-Bit Server VM version 24.80-b11

JDK 7.0\_80

Alternatively, you can use ps to find out which java binary was used:

ps aux | grep 9619

user 9619 0.0 0.5 4031224 82480 ? Sl 12:31 0:01 /usr/lib/jvm/java-7-oracle/bin/java -cp /usr/lib/jvm

--

Destination Folder: C:\oraclexe\

Port for 'Oracle Database Listener': 1521

Port for 'Oracle Services for Microsoft Transaction Server': 2030

Port for HTTP Listener: 8080

Tom cat port changed to 8085

https://github.com/Dotridge/springcloudrepo

Description Resource Path Location Type

The container 'Maven Dependencies' references non existing library 'C:\Users\NARESH\.m2\repository\joda-time\joda-time\2.9.2\joda-time-2.9.2.jar' gs-maven-complete Build path Build Path Problem

Login with User Sam & password abc125, check RememberMe as well.

<http://websystique.com/springmvc/spring-mvc-4-and-spring-security-4-integration-example/>

like this all typr of xml desc for servlets 3,3.1,2.5,.2.4,… link:

https://www.mkyong.com/web-development/the-web-xml-deployment-descriptor-examples/

<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee

http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"

version="3.1">

</web-app>

Maven like technical kt quick useful:

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.7.0</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>

--

<dependency>

<groupId>com.oracle</groupId>

<artifactId>ojdbc14</artifactId>

<version>10.2.3.0</version>

<scope>system</scope>

<systemPath>C:/oraclexe/app/oracle/product/10.2.0/server/jdbc/lib/ojdbc14.jar</systemPath>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>5.1.31</version>

</dependency>

<dependency>

<groupId>com.oracle</groupId>

<artifactId>ojdbc14</artifactId>

<version>10.1</version>

</dependency>

<dependency>

<groupId>ojdbc14</groupId>

<artifactId>ojdbc14</artifactId>

<version>10.2.0.3.0</version>

</dependency>

<bean id=*"ds"* class=*"org.springframework.jdbc.datasource.DriverManagerDataSource"*>

<property name=*"driverClassName"* value=*"com.mysql.jdbc.Driver"* />

<property name=*"url"* value=*"jdbc:mysql://localhost:3306/db2"* />

<property name=*"username"* value=*"root"* />

<property name=*"password"* value=*"root"* />

</bean>

spring.datasource.driver-class-name= *com.mysql.jdbc.Driver*

spring.datasource.url= *jdbc:mysql://localhost:3306/cms* spring.datasource.username=root

spring.datasource.password=root

DROP TABLE IF EXISTS t1, t2;

CREATE TABLE t1 (

    id INT PRIMARY KEY AUTO\_INCREMENT

);

CREATE TABLE t2 (

    id VARCHAR(20) PRIMARY KEY,

ref INT NOT NULL

);

INSERT INTO t1 VALUES (1),(2),(3);

INSERT INTO t2(ref) VALUES('A',1),('B',2),('C',3);

<http://www.java67.com/2015/06/how-to-fix-javalangclassnotfoundexception-oracle.jdbc.driver.oracledriver.html>

\*\*\*\*\*hibernate-configuration(hibernate-cfg.xml)

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<property name=*"hibernate.connection.driver\_class"*>com.mysql.jdbc.Driver</property>

<property name=*"hibernate.connection.url"*>jdbc:mysql://localhost:3306/db2</property>

<property name=*"hibernate.connection.username"*>root</property>

<property name=*"hibernate.connection.password"*>root</property>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.MySQLDialect</property>

<property name=*"show\_sql"*>true</property>

<property name=*"format\_sql"*>false</property>

<mapping resource=*"Products.hbm.xml"* />

</session-factory>

</hibernate-configuration>

persistence.xml

<?xml version="1.0" encoding="UTF-8"?>

<persistence version="2.1"

xmlns="http://xmlns.jcp.org/xml/ns/persistence"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence\_2\_1.xsd">

<persistence-unit name="myPU" transaction-type="JTA">

<properties>

<property name="javax.persistence.schema-generation.database.action" value="drop-and-create"/>

<property name="javax.persistence.schema-generation.create-source" value="script"/>

<property name="javax.persistence.schema-generation.drop-source" value="script"/>

<property name="javax.persistence.schema-generation.create-script-source" value="sql/create.sql"/>

<property name="javax.persistence.schema-generation.drop-script-source" value="sql/drop.sql"/>

<property name="javax.persistence.sql-load-script-source" value="sql/load.sql"/>

</properties>

</persistence-unit>

</persistence>

<prop key=*"hibernate.dialect"*>org.hibernate.dialect.MySQLDialect</prop>

<bean id="ds class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName" value="oracle.jdbc.driver.OracleDriver" />

<property name="url" value="jdbc:oracle:thin:@localhost:1521/XE" />

<property name="username" value="system" />

<property name="password" value="admin" />

</bean>

<dependency>

<groupId>com.oracle</groupId>

<artifactId>ojdbc14</artifactId>

<version>10.1</version>

</dependency>

**Table 5.1 Connection Properties - Miscellaneous**

| **These MySQL Data Types** | **Can always be converted to these Java types** |
| --- | --- |
| CHAR, VARCHAR, BLOB, TEXT, ENUM, and SET | java.lang.String, java.io.InputStream, java.io.Reader, java.sql.Blob, java.sql.Clob |
| FLOAT, REAL, DOUBLE PRECISION, NUMERIC, DECIMAL, TINYINT, SMALLINT, MEDIUMINT, INTEGER, BIGINT | java.lang.String, java.lang.Short, java.lang.Integer, java.lang.Long, java.lang.Double, java.math.BigDecimal |
| DATE, TIME, DATETIME, TIMESTAMP | java.lang.String, java.sql.Date, java.sql.Timestamp |

**Table 5.2 MySQL Types to Java Types for ResultSet.getObject()**

| **MySQL Type Name** | **Return value of**GetColumnClassName | **Returned as Java Class** |
| --- | --- | --- |
| BIT(1) (new in MySQL-5.0) | BIT | java.lang.Boolean |
| BIT( > 1) (new in MySQL-5.0) | BIT | byte[] |
| TINYINT | TINYINT | java.lang.Boolean if the configuration property tinyInt1isBit is set totrue (the default) and the storage size is 1, or java.lang.Integer if not. |
| BOOL, BOOLEAN | TINYINT | See TINYINT, above as these are aliases for TINYINT(1), currently. |
| SMALLINT[(M)] [UNSIGNED] | SMALLINT [UNSIGNED] | java.lang.Integer (regardless of whether it is UNSIGNED or not) |
| MEDIUMINT[(M)] [UNSIGNED] | MEDIUMINT [UNSIGNED] | java.lang.Integer (regardless of whether it is UNSIGNED or not) |
| INT,INTEGER[(M)] [UNSIGNED] | INTEGER [UNSIGNED] | java.lang.Integer, if UNSIGNED java.lang.Long |
| BIGINT[(M)] [UNSIGNED] | BIGINT [UNSIGNED] | java.lang.Long, if UNSIGNED java.math.BigInteger |
| FLOAT[(M,D)] | FLOAT | java.lang.Float |
| DOUBLE[(M,B)] | DOUBLE | java.lang.Double |
| DECIMAL[(M[,D])] | DECIMAL | java.math.BigDecimal |
| DATE | DATE | java.sql.Date |
| DATETIME | DATETIME | java.sql.Timestamp |
| TIMESTAMP[(M)] | TIMESTAMP | java.sql.Timestamp |
| TIME | TIME | java.sql.Time |
| YEAR[(2|4)] | YEAR | If yearIsDateType configuration property is set to false, then the returned object type is java.sql.Short. If set to true (the default), then the returned object is of type java.sql.Date with the date set to January 1st, at midnight. |
| CHAR(M) | CHAR | java.lang.String (unless the character set for the column is BINARY, thenbyte[] is returned. |
| VARCHAR(M) [BINARY] | VARCHAR | java.lang.String (unless the character set for the column is BINARY, thenbyte[] is returned. |
| BINARY(M) | BINARY | byte[] |
| VARBINARY(M) | VARBINARY | byte[] |
| TINYBLOB | TINYBLOB | byte[] |
| TINYTEXT | VARCHAR | java.lang.String |
| BLOB | BLOB | byte[] |
| TEXT | VARCHAR | java.lang.String |
| MEDIUMBLOB | MEDIUMBLOB | byte[] |
| MEDIUMTEXT | VARCHAR | java.lang.String |
| LONGBLOB | LONGBLOB | byte[] |
| LONGTEXT | VARCHAR | java.lang.String |
| ENUM('value1','value2',...) | CHAR | java.lang.String |
| SET('value1','value2',...) | CHAR | java.lang.String |

hibernate-configuration(hibernate-cfg.xml)

<hibernate-configuration>

<session-factory>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.MySQLDialect</property>

<property name=*"hibernate.connection.driver\_class"*>com.mysql.jdbc.Driver</property>

<property name=*"hibernate.connection.username"*>root</property>

<property name=*"hibernate.connection.password"*>root</property>

<property name=*"hibernate.connection.url"*>jdbc:mysql://localhost:3306/mkyong</property>

<property name=*"show\_sql"*>true</property>

<property name=*"format\_sql"*>false</property>

<mapping resource=*"com/websystique/hibernate/model/Student.hbm.xml"*></mapping>

</session-factory>

</hibernate-configuration>

--

Hbm,xml

<?xml version=*"1.0"* encoding=*"utf-8"*?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD//EN" "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<hibernate-mapping>

<class name=*"com.websystique.hibernate.model.Student"* table=*"STUDENT"*>

<id name=*"id"* type=*"int"* column=*"id"*>

<generator class=*"native"* />

</id>

<property name=*"firstName"* column=*"FIRST\_NAME"* type=*"string"* />

<property name=*"lastName"* column=*"LAST\_NAME"* type=*"string"* />

<property name=*"section"* column=*"SECTION"* type=*"string"* />

</class>

</hibernate-mapping>

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<property name=*"hibernate.connection.driver\_class"*>com.mysql.jdbc.Driver</property>

<property name=*"hibernate.connection.url"*>jdbc:mysql://localhost:3306/db2</property>

<property name=*"hibernate.connection.username"*>root</property>

<property name=*"hibernate.connection.password"*>root</property>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.MySQLDialect</property>

<property name=*"show\_sql"*>true</property>

<property name=*"format\_sql"*>false</property>

<mapping resource=*"Products.hbm.xml"* />

</session-factory>

</hibernate-configuration>

servlet-context.xml code:

<?xml version="1.0" encoding="UTF-8"?>

<beans:beans xmlns="http://www.springframework.org/schema/mvc"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:beans="http://www.springframework.org/schema/beans"

xmlns:context="http://www.springframework.org/schema/context"

xsi:schemaLocation="http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc.xsd

http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd">

<!-- DispatcherServlet Context: defines this servlet's request-processing

infrastructure -->

<!-- Enables the Spring MVC @Controller programming model -->

<annotation-driven />

<!-- Handles HTTP GET requests for /resources/\*\* by efficiently serving

up static resources in the ${webappRoot}/resources directory -->

<resources mapping="/\*\*" location="/" />

<!-- Resolves views selected for rendering by @Controllers to .jsp resources

in the /WEB-INF/views directory -->

<beans:bean

class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<beans:property name="prefix" value="/WEB-INF/views/" />

<beans:property name="suffix" value=".jsp" />

</beans:bean>

<beans:bean id="multipartResolver"

class="org.springframework.web.multipart.commons.CommonsMultipartResolver">